

Throughout this document  $x$  and  $y$  will be either row or column vectors and  $A$  will always be a matrix.

| Basics                           |                             |
|----------------------------------|-----------------------------|
| <code>clc</code>                 | Clear command window        |
| <code>clear</code>               | Clear all variables         |
| <code>clf</code>                 | Clear all plots             |
| <code>close all</code>           | Close all plots             |
| <code>doc function</code>        | Open help page for function |
| <code>% This is a comment</code> | Comments                    |
| <code>ctrl-c</code>              | Abort the current operation |
| <code>format short</code>        | Display 4 decimal places    |
| <code>format long</code>         | Display 15 decimal places   |
| <code>edit filename</code>       | Opens filename in editor    |
| <code>disp('text')</code>        | Print text                  |

| Keyboard Shortcuts |                     |                                |
|--------------------|---------------------|--------------------------------|
| Win                | Mac                 | info                           |
| F1                 | F1                  | docs for highlighted function  |
| F5                 | ⌘+ R                | Run code                       |
| F9                 | ⌘+ ↵                | Run selected code              |
| F11                | F11                 | Run code line, enter functions |
| Shift + F5         | Shift + F5          | Leave debugger                 |
| F12                | ⌘+ \                | Insert break point             |
| Ctrl + Pg Up/Down  | Ctrl + Fn + up/down | Moves between tabs             |
| Ctrl + shift       | Ctrl + shift        | Moves between components       |
| Ctrl + C           | Ctrl + C            | Interrupts code                |
| Ctrl + D           | Shift +⌘+ D         | Open highlighted codes file    |
| Ctrl + R/T         | ⌘+ / ⌘+ T           | Comment/uncomment              |
| Ctrl + N           | Ctrl + N            | New script                     |
| Ctrl + W           | Ctrl + W            | Close script                   |
| Ctrl + shift + d   | Ctrl + shift + d    | Docks window                   |
| Ctrl + shift + u   | Ctrl + shift + u    | Undocks window                 |
| Ctrl + shift + m   | Ctrl + shift + m    | max window                     |

| Defining and Changing Variables                                  |   |
|--|---|
| <code>a = 3</code>   | Define variable $a$ to be 3                   |
| <code>x = [1, 2, 3]</code>                                       | Set $x$ to be the row vector $[1, 2, 3]$      |
| <code>x = [1; 2; 3]</code>                                       | Set $x$ to be the column vector $[1, 2, 3]^T$ |
| <code>A = [1, 2, 3, 4;<br/>5, 6, 7, 8;<br/>9, 10, 11, 12]</code> | Set $A$ to be a $3 \times 4$ matrix           |
| <code>x(2) = 7</code>  | Change $x$ from $[1, 2, 3]$ to $[1, 7, 3]$    |
| <code>A(2,1) = 0</code>  | Change $A_{2,1}$ from 5 to 0                  |

| Basic Arithmetic and Functions  |                                    |
|---------------------------------|------------------------------------|
| <code>3*4, 7+4, 2-6, 8/3</code> | multiply, add, subtract and divide |
| <code>3^7</code>                | Compute $3^7$                      |
| <code>sqrt(5)</code>            | Compute $\sqrt{5}$                 |
| <code>log(3)</code>             | Compute $\ln(3)$                   |
| <code>log10(100)</code>         | Compute $\log_{10}(100)$           |
| <code>abs(-5)</code>            | Compute $ -5 $                     |
| <code>sin(5*pi/3)</code>        | Compute $\sin(5\pi/3)$             |
| <code>floor(3.8)</code>         | Compute $\lfloor 3.8 \rfloor$      |

| Constructing Matrices and Vectors     |  |
|---------------------------------------|--|
| <code>zeros(12, 5)</code>             | Make a $12 \times 5$ matrix of zeros   |
| <code>ones(12, 5)</code>              | Make a $12 \times 5$ matrix of ones  |
| <code>eye(5)</code>                   | Make a $5 \times 5$ identity matrix  |
| <code>eye(12, 5)</code>               | Make a $12 \times 5$ identity matrix   |
| <code>linspace(1.4, 6.3, 1004)</code> | Make a vector with 1004 elements evenly spaced between 1.4 and 6.3                                     |
| <code>logspace(1.4, 6.3, 1004)</code> | Make a vector with 1004 elements where the log of the spacing is evenly increasing between 1.4 and 6.3 |
| <code>7:15</code>                     | Row vector of 7, 8, $\dots$ , 14, 15   |

## Operations on Matrices and Vectors

|                      |   |
|----------------------|---|
| <code>3 * x</code>   | Multiply every element of $x$ by 3                  |
| <code>x + 2</code>   | Add 2 to every element of $x$                       |
| <code>x + y</code>   | Element-wise addition of two vectors $x$ and $y$    |
| <code>A * y</code>   | Product of a matrix and vector                      |
| <code>A * B</code>   | Product of two matrices                             |
| <code>A .* B</code>  | Element-wise product of two matrices                |
| <code>A ^ 3</code>   | Square matrix $A$ to the third power                |
| <code>A .^ 3</code>  | Every element of $A$ to the third power             |
| <code>cos(A)</code>  | Compute the cosine of every element of $A$          |
| <code>abs(A)</code>  | Compute the absolute values of every element of $A$ |
| <code>A'</code>      | Transpose of $A$                                    |
| <code>inv(A)</code>  | Compute the inverse of $A$                          |
| <code>det(A)</code>  | Compute the determinant of $A$                      |
| <code>eig(A)</code>  | Compute the eigenvalues of $A$                      |
| <code>size(A)</code> | Get the size of $A$                                 |

## Entries of Matrices and Vectors

|                         |   |
|-------------------------|---|
| <code>x(2:12)</code>    | The 2 <sup>nd</sup> to the 12 <sup>th</sup> elements of $x$ |
| <code>x(2:end)</code>   | The 2 <sup>nd</sup> to the last elements of $x$             |
| <code>x(1:3:end)</code> | Every third element of $x$ from the first to last           |
| <code>A(5,:)</code>     | Get the 5 <sup>th</sup> row of $A$                          |
| <code>A(:,5)</code>     | Get the 5 <sup>th</sup> column of $A$                       |
| <code>A(5, 1:3)</code>  | Get the first to third elements in the 5 <sup>th</sup> row  |

## Plotting

|                                   |   |
|-----------------------------------|---|
| <code>plot(x,y)</code>            | Plot $y$ versus $x$ (must be the same length)                               |
| <code>loglog(x,y)</code>          | Plot $y$ versus $x$ on a log-log scale (both axes have a logarithmic scale) |
| <code>semilogx(x, y)</code>       | Plot $y$ versus $x$ with $x$ on a log scale                                 |
| <code>semilogy(x, y)</code>       | Plot $y$ versus $x$ with $y$ on a log scale                                 |
| <code>axis equal</code>           | Force the $x$ and $y$ axes to be scaled equally                             |
| <code>title('A Title')</code>     | Add a title to the plot   |
| <code>xlabel('x label')</code>    | Add a label to the $x$ axis   |
| <code>ylabel('y label')</code>    | Add a label to the $y$ axis   |
| <code>legend('foo', 'bar')</code> | Label 2 curves for the plot   |
| <code>grid</code>                 | Add a grid to the plot  |
| <code>hold on</code>              | Multiple plots on single figure   |
| <code>figure</code>               | Start a new plot  |

## Constants

|                      |  |
|----------------------|--|
| <code>pi</code>      | $\pi = 3.141592653589793$  |
| <code>NaN</code>     | Not a number (i.e. 0/0)  |
| <code>Inf</code>     | Infinity   |
| <code>realmax</code> | Largest positive floating-point number $1.7977 \cdot 10^{308}$   |
| <code>realmin</code> | Smallest positive floating-point number $2.2251 \cdot 10^{-308}$ |

## Saving and loading files

|                              |   |
|------------------------------|---|
| <code>save myfile.mat</code> | Saves workspace variables in <code>myfile.mat</code>                  |
| <code>load myfile.mat</code> | Loads variables in <code>myfile.mat</code> into the current workspace |

## Existence of names

|                              |   |
|------------------------------|---|
| <code>exist some_name</code> | Returns numeric code, depending on whether this name already exists and if so, also depending on it's category (built-in, user-defined, etc.) |
|------------------------------|---|

## Matrix and array commands

|                      |                                   |
|----------------------|-----------------------------------|
| <code>repelem</code> | Repeat copies of array elements   |
| <code>repmat</code>  | Repeat copies of array            |
| <code>repmat</code>  | Repeat copies of array            |
| <code>length</code>  | Length of largest array dimension |
| <code>sort</code>    | sort array elements               |
| <code>flip</code>    | flip order of elements            |
| <code>reshape</code> | Reshape array                     |
| <code>end</code>     | Denotes last element              |
| <code>size(A)</code> | Get the size of $A$               |

## Linear algebra

|                                |   |
|--------------------------------|---|
| <code>x = A\b</code>           | Solves $Ax = b$   |
| <code>x = linsolve(A,B)</code> | Solves $A \cdot X = B$  |
| <code>[V,D] = eig(A,B)</code>  | Diagonal matrix $D$ of eigenvalues and matrix $V$ whose columns are the corresponding right eigenvectors, so that $A \cdot V = V \cdot D$ . |
| <code>[U,S,V] = svd(A)</code>  | Singular value decomposition of matrix $A$ , such that $A = U \cdot S \cdot V'$ .   |
| <code>inv</code>               | Matrix inverse  |
| <code>realmin</code>           | Smallest positive floating-point number $2.2251 \cdot 10^{-308}$  |

## For loops

```
for k = 1:5
    disp(k);
end
```

## While loops

```
k = 0;
while k < 7
    k = k + 1;
end
```

## Logicals

```
a = 10; % Assign a the value of 10
a == 5 % Test if a is equal to 5
    false
a == 10 % Test if a is equal to 10
    true
a >= 5 % Test if a is greater than or equal to 5
    true
a < 11 % Test if a is less than 11
    true
a ~= 4 % Test if a is not equal to 4
    true
a > 1 && a ~= 10 % Test if a is greater than 1 AND
    false % not equal to 10
a > 1 || a ~= 10 % Test if a is greater than 1 OR
    true % not equal to 10
```

## Conditional Statements

```
if a > 10
    disp('Greater than 10');
elseif a == 5
    disp('a is 5');
else
    disp('Neither condition met');
end
```

## Functions

```
function output = addNumbers(x, y)
    output = x + y;
end

addNumbers(10, -5)
5
```

## Function Handles

```
f = @ (x) sin(x.^2)./(5*x);

f(pi/2)
0.0795
f([-pi/2, 0, pi/2])
-0.0795 NaN 0.0795
```

## Plotting

```
x = linspace(-3*pi, 3*pi, 1000);
y1 = sin(x);
y2 = cos(x);

plot(x, y1, 'k-'); % Plot sin(x) as a black line
hold on           % Now we can add another curve
plot(x, y2, 'r-'); % Plot cos(x) as a red line

% Set the axis limits
axis([-3*pi, 3*pi, -1.5, 1.5])

% Add axis labels
xlabel('x');
ylabel('y');

% Add a title
title('A plot of cos(x) and sin(x)');

% Add a legend
legend('sin(x)', 'cos(x)');
```

